

ABSTRACT

A simultaneous and redundantly threaded, pipelined processor executes the same set of instructions simultaneously as two separate threads to provide fault tolerance. One thread is processed ahead of the other thread so that the instructions in one thread are processed through the processor's pipeline ahead of the corresponding instructions from the other thread. The thread, whose instructions are processed earlier, places its committed stores in a store queue. Subsequently, the second thread places its committed stores in the store queue. A compare circuit periodically scans the store queue for matching store instructions. If otherwise matching store instructions differ in any way (address or data), then a fault has occurred in the processing and the compare circuits initiates fault recovery. If comparison of the two instructions reveals they are identical, the compare circuit allows only a single store instruction to pass to the data cache or the system main memory. In this way, transient faults are detected with a minimum amount of hardware overhead and independent of differences in the actual order of program execution or differences in branch speculation.